Fair task allocation in transportation

Qing Chuan Ye a Yingqian Zhang b Rommert Dekker a

a Erasmus University Rotterdam, P.O. Box 1738, 3000 DR Rotterdam
b Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven

Overview In [1], we study a task allocation problem in which we take fairness into account in addition to the standard minimum cost criterion. This work was inspired by an actual transportation situation in the port of Rotterdam. Due to an increase in inter-terminal transport, an idea was proposed to use already present trucks to execute open jobs, in which daily repeated auctions will be used. In this task allocation problem, the port tries to give jobs to each company according to their availability and their cost for each job. Jobs have designated time slots and can only be allocated to one bidder, while bidders have a limited number of trucks available per time slot. Due to the repetitive nature, we do not only look at optimizing the costs in the task allocation, but we also incorporate fairness in the task allocation. For the fairness criterion we use the notion of max-min fairness. The max-min fairness principle entails that given a total number of jobs, the number of jobs for any company cannot be increased by at the same time decreasing the number of jobs of the other companies that have the same number of jobs or less. Intuitively speaking, we want to allocate jobs among the companies as evenly as possible.

We formulate this problem as a “max-min fair minimum cost allocation problem” (MFMCA), in which we aim for a polynomial-time solution. The difficulty of our problem lies in the additional fairness criterion, which requires the developed algorithm to satisfy three criteria: allocation maximization, fairness, and cost minimization. To the best of our knowledge, no existing polynomial-time algorithm can be directly applied to solve our problem. In [1], we propose polynomial-time algorithms to solve MFMCA as a two-level optimization problem. First, we aim at a fairest allocation among companies while ensuring that a maximal set of tasks can be allocated for execution. We call this the “max-min fair allocation problem” (MMFA). Second, because there might be an exponential number of allocations that are considered max-min fair, we would like to determine which of these fair allocations has the lowest cost. The resulting allocation is max-min fair with minimum cost. Using computational results, we provide insights into situations in which fairness can be incorporated without giving up too much efficiency.

Algorithms First, to solve MMFA, we propose an algorithm called IMaxFlow, which iteratively applies a maximum flow algorithm in a greedy fashion. In this way, the flow assigned to each company is increased step by step until no more flows can be assigned. Given an instance of the MMFA problem, we can construct a flow network and apply IMaxFlow to obtain the optimal max-min fairness vector, which specifies the fairest task distribution among agents given a maximal set of tasks that can be allocated.

Once we know the fairness vector from IMaxFlow, we want to minimize the associated cost. However, we cannot apply a standard minimum-cost maximum-flow algorithm to our flow network as it may violate the max-min fairness condition while looking for the minimum cost. The obtained fairness vector tells us in what quantities the jobs will be distributed in the fairest allocation. However, we do not know which company would be assigned which number of jobs such that the total cost is smallest. Therefore, we propose the FairMinCost algorithm, which adjusts the flow network in such a way that a standard minimum-cost maximum-flow algorithm simultaneously decides which company is assigned which quantity from the fairness vector, and thus will not violate the max-min fairness condition. This is done by introducing a set of dummy jobs and corresponding additional dummy nodes, dependent on the

1The full paper has been accepted in Omega, 2016 [1].
optimal max-min fairness vector, to the flow network. We prove in [1] that the solution obtained by applying any standard minimum-cost maximum-flow algorithm on this altered flow network is equivalent to the optimal solution to MFMCA.

**Experiments** We investigate the performance of the proposed algorithms compared with an optimal algorithm for the task allocation problem with a sole objective of minimum cost through numerical experiments. We are interested in two performance measurements: (1) the price of fairness (POF), i.e., the relative increase in the total cost under the fair solution with regard to the minimum-cost solution; (2) the effect of fairness on the job distribution, i.e., the number of jobs assigned to each company. We look at several scenarios for both the jobs that are being auctioned (homogeneous and heterogeneous costs), and the companies who are bidding on the jobs (low, high or mixed competition). We also look at different capacities for the companies (5% and 10% capacity). The experimental results show that the job allocation in the minimum-cost solution is very skewed, where some companies obtain many jobs, while others end up with none. The fair allocation allocates the jobs much more evenly over the companies (see Figure 1). However, this of course comes at a cost (see Figure 2). In general, when the costs among companies are similar (homogeneous cost), implementing fair allocations comes with almost no extra cost for the task owner. When the prices are highly volatile however (heterogeneous cost), the auctioneer may need to pay more for the fairness. In additional experiments, we observe that when there are relatively few jobs, the POF will usually be relatively high due to lack of flexibility in reallocating jobs, but it stabilizes as the number of jobs increases. Similarly, the more companies are participating, the more bids there will be, resulting in more flexibility for reallocation and a lower POF. When there are only few companies, there is lack of flexibility for reallocation, resulting in a fair allocation similar to the minimum-cost allocation. This means that the number of participants should be sufficient in order to have the desired flexibility needed for reallocation. All in all, among the majority of test instances, fairness in the job distribution comes with a very small price in terms of cost. Although the additional fairness criterion introduces a problem that cannot be solved by known standard algorithms, our proposed algorithms solve MFMCA in polynomial time. Furthermore, our algorithms can be applied to many other task allocation problem in which the centralized planner wants to enforce some kind of fairness among the agents. This is especially relevant in the upcoming sharing economy and collaborative consumption in transport, e.g., car-sharing.

**References**